

Stock Market Forecasting Using Hidden Markov Model: A New Approach

Md. Rafiul Hassan and Baikunth Nath

Computer Science and Software Engineering
The University of Melbourne, Carlton 3010, Australia.
{mrhassan , bnath}@cs.mu.oz.au

Abstract

This paper presents Hidden Markov Models (HMM) approach for forecasting stock price for interrelated markets. We apply HMM to forecast some of the airlines stock. HMMs have been extensively used for pattern recognition and classification problems because of its proven suitability for modelling dynamic systems. However, using HMM for predicting future events is not straightforward. Here we use only one HMM that is trained on the past dataset of the chosen airlines. The trained HMM is used to search for the variable of interest behavioural data pattern from the past dataset. By interpolating the neighbouring values of these datasets forecasts are prepared. The results obtained using HMM are encouraging and HMM offers a new paradigm for stock market forecasting, an area that has been of much research interest lately.

Key Words: HMM, stock market forecasting, financial time series, feature selection

1. Introduction

Forecasting stock price or financial markets has been one of the biggest challenges to the AI community. The objective of forecasting research has been largely beyond the capability of traditional AI research which has mainly focused on developing intelligent systems that are supposed to emulate human intelligence. By its nature the stock market is mostly complex (non-linear) and volatile. The rate of price fluctuations in such series depends on

many factors, namely equity, interest rate, securities, options, warrants, merger and ownership of large financial corporations or companies etc. Human traders can not consistently win in such markets. Therefore, developing AI systems for this kind of forecasting requires an iterative process of knowledge discovery and system improvement through data mining, knowledge engineering, theoretical and data-driven modelling, as well as trial and error experimentation.

The stock markets in the recent past have become an integral part of the global economy. Any fluctuation in this market influences our personal and corporate financial lives, and the economic health of a country. The stock market has always been one of the most popular investments due to its high returns [1]. However, there is always some risk to investment in the Stock market due to its unpredictable behaviour. So, an 'intelligent' prediction model for stock market forecasting would be highly desirable and would of wider interest.

A large amount of research has been published in recent times and is continuing to find an optimal (or nearly optimal) prediction model for the stock market. Most of the forecasting research has employed the statistical time series analysis techniques like auto-regression moving average (ARMA) [2] as well as the multiple regression models. In recent years, numerous stock prediction systems based on AI techniques, including artificial neural networks (ANN) [3, 4, 5], fuzzy logic [6], hybridization of ANN and fuzzy system [7, 8, 9], support vector machines [10] have been proposed. However, most of them have their own constraints. For instance, ANN is very much problem oriented because of its chosen architecture. Some researchers have used fuzzy systems to develop a model to forecast stock market behaviour. To build a fuzzy system one requires some background expert knowledge.

In this paper, we make use of the well established Hidden Markov Model (HMM) technique to forecast stock price for some of the airlines. The HMMs have been extensively used in the area like speech recognition, DNA

sequencing, electrical signal prediction and image processing, etc. In here, HMM is used in a new way to develop forecasts. First we locate pattern(s) from the past datasets that match with today's stock price behaviour, then interpolate these two datasets with appropriate neighbouring price elements and forecast tomorrow's stock price of the variable of interest. Details of the proposed method are provided in Section 3. The remainder of the paper is organised as follows: Section 2 provides a brief overview on HMM; Section 4 lists experimental results obtained using HMM and compares with results obtained using ANN; and finally Section 5 concludes the paper.

2. HMM as a Predictor

A Hidden Markov Model (HMM) is a finite state machine which has some fixed number of states. It provides a probabilistic framework for modelling a time series of multivariate observations. Hidden Markov models were introduced in the beginning of the 1970's as a tool in speech recognition. This model based on statistical methods has become increasingly popular in the last several years due to its strong mathematical structure and theoretical basis for use in a wide range of applications. In recent years researchers proposed HMM as a classifier or predictor for speech signal recognition [11, 12, 13], DNA sequence analysis [14], handwritten characters recognition [15], natural language domains etc. It is clear that HMM is a very powerful tool for various applications. The advantage of HMM can be summarized as:

- HMM has strong statistical foundation
- It is able to handle new data robustly
- Computationally efficient to develop and evaluate (due to the existence of established training algorithms).
- It is able to predict similar patterns efficiently [16]

Rabiner [17] tutorial explains the basics of HMM and how it can be used for signal prediction. The next session describes the HMM in brief.

2.1. The Hidden Markov Model

Hidden Markov Model is characterized by the following

- 1) number of states in the model
- 2) number of observation symbols
- 3) state transition probabilities
- 4) observation emission probability distribution that characterizes each state
- 5) initial state distribution

For the rest of this paper the following notations will be used regarding HMM

N = number of states in the model

M = number of distinct observation symbols per state (observation symbols correspond to the physical output of the system being modelled)

T = length of observation sequence

O = observation sequence, i.e., $O_1, O_2, O_3, \dots, O_T$

Q = state sequence q_1, q_2, \dots, q_T in the Markov model

$A = \{a_{ij}\}$ transition matrix, where a_{ij} represents the transition probability from state i to state j

$B = \{b_i(O_t)\}$ observation emission matrix, where $b_i(O_t)$ represent the probability of observing O_t at state j

$\pi = \{\pi_i\}$ the prior probability, where π_i represent the probability of being in state i at the beginning of the experiment, i.e., at time $t = 1$

$\lambda = (A, B, \pi)$ the overall HMM model.

As mentioned above the HMM is characterized by N, M, A, B and π . The $a_{ij}, b_i(O_t)$, and π_i have the properties

$$\sum_j a_{ij} = 1, \quad \sum_t b_i(O_t) = 1, \quad \sum_i \pi_i = 1 \text{ and}$$

$$a_{ij}, b_i(O_t), \pi_i \geq 0 \text{ for all } i, j, t.$$

To work with HMM, the following three fundamental questions should be resolved

1. Given the model $\lambda = (A, B, \pi)$ how do we compute $P(O|\lambda)$, the probability of occurrence of the observation sequence $O = O_1, O_2, \dots, O_T$.
2. Given the observation sequence O and a model λ , how do we choose a state sequence q_1, q_2, \dots, q_T that best explains the observations.
3. Given the observation sequence O and a space of models found by varying the model parameters A, B and π , how do we find the model that best explains the observed data.

There are established algorithms to solve the above questions. In our task we have used the forward-backward algorithm to compute the $P(O|\lambda)$, Viterbi algorithm to resolve problem 2, and Baum-Welch algorithm to train the HMM. The details of these algorithms are given in the tutorial by Rabiner [17].

3. Using HMM for Stock market forecasting

In this section we develop an HMM based tool for time series forecasting, for instance for the stock market

forecasting. While implementing the HMM, the choice of the model, choice of the number of states and observation symbol (continuous or discrete or multi-mixture) become a tedious task. For instance we have used left-right HMM with 4 states.

In our problem, for simplicity, we consider 4 input features for a stock – that is the opening price, closing price, highest price, and the lowest price. The next day's closing price is taken as the target price associated with the four input features. Our observations here being continuous rather than discrete, we choose empirically as many as 3 mixtures for each state for the model density.

For the prior probability π_i , a random number was chosen and normalized so that $\sum_{i=1}^N \pi_i = 1$. The dataset

being continuous, the probability of emitting symbols from a state can not be calculated. For this reason a three-dimensional Gaussian distribution was initially chosen as the observation probability density function. Thus, we have

$$b_j(O) = \sum c_{jm} \mathcal{N}[O, \mu_{jm}, U_{jm}] \quad 1 \leq j \leq N$$

where

O = vector of observations being modelled

c_{jm} = mixture coeff. for the m-th mixture in state j ,

where $\sum_{m=1}^M c_{jm} = 1$

μ_{jm} = mean vector for the m-th mixture component in state j

U_{jm} = Covariance matrix for the m-th mixture component in state j

\mathcal{N} = Gaussian density.

In the experiment, our objective was to predict the next day's closing price for a specific stock market share using aforementioned HMM model. For training the model, past one and a half years (approximately) daily data were used and recent last three month's data were used to test the efficiency of the model. The input and output data features were as follows:

Input: opening, high, low, and closing price

Output: next day's closing price

The idea behind our new approach in using HMM is that of using the training dataset for estimating the parameter set (A, B, π) of the HMM. For a specific stock at the market close we know day's price values of the four variables (open, high, low, close), and using this information our objective is to predict next day's closing

price. A forecast of any of the four variables for the next day indeed will be of tremendous value to the traders and investors. Using the trained HMM, likelihood value for current day's dataset is calculated. For example, say the likelihood value for the day is 'f', then from the past dataset using the HMM we locate those instances that would produce the same 'f' or nearest to the 'f' likelihood value. That is we locate the past day(s) where the stock behaviour is similar to that of the current day. Assuming that the next day's stock price should follow about the same past data pattern, from the located past day(s) we simply calculate the difference of that day's closing price and next to that day's closing price. Thus the next day's stock closing price forecast is established by adding the above difference to the current day's closing price.

4. Experimentation: Training and Testing

In order to train the HMM, we divided the dataset into two sets, one training set and one test (recall) set. For example, we trained an HMM using the daily stock data of Southwest Airlines for the period 18 December 2002 to 29 September 2004 to predict the closing price on 30 September 2004. The trained HMM produced likelihood value of -9.4594 for the stock price on 29 September 2004. Using this trained HMM and the past data, we located a (closer) likelihood value -9.4544 on 01 July 2003. Figure 1 shows the similarities between these two datasets (stock prices on 30 September 2004 and 01 July 2003). It seems quite logical that 29 September 2004 stock behaviour should follow the behaviour that of 01 July 2003.

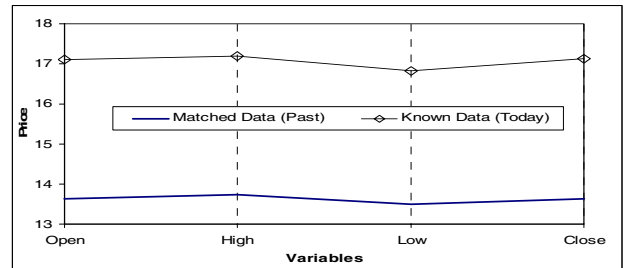


Figure 1. The current day's stock price behaviour matched with past day's price data

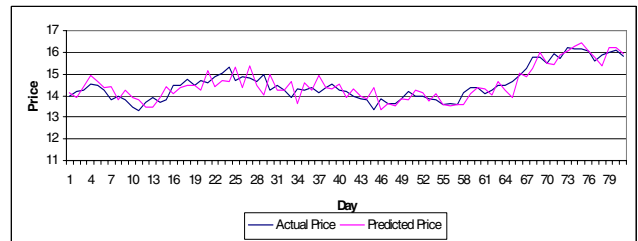


Figure 2. Actual Vs. Predicted stock price

Table 1. Dataset along with the matched past dataset

	Opening price	High price	Low price	Closing price	Predicted closing price (30 Sep 2004)	Actual closing price (30 Sep 2004)
Today's data 29 Sep 2004	\$13.63	\$13.73	\$13.49	\$13.62	\$13.85	\$13.85
Matched data pattern using HMM 01 Jul 2003	\$17.1	\$17.2	\$16.83	\$17.13		
Next day's data 02 Jul 2003				\$17.36		

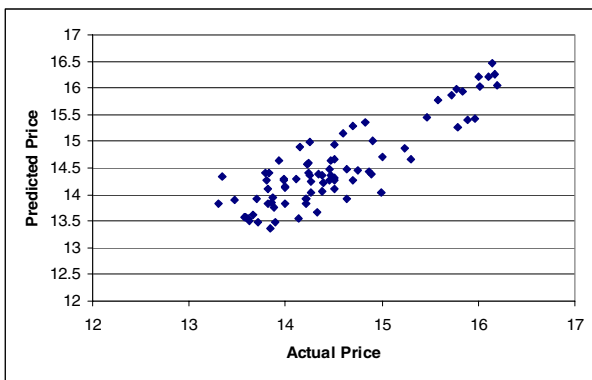


Figure 3. The correlation between predicted and actual closing stock price

We, therefore, calculated the difference between the closing prices on 01 July 2003 and the next day 02 July 2003. That is $\$17.36 - \$17.13 = \$0.23$. Then this difference is added to the closing price on 29 September 2004 to forecast closing price for 30 September 2004. Table 1 shows the predicted and the actual prices of stock on 30 September 2004. Figure 2 shows the prediction accuracy of the model and Figure 3 shows the correlation among the predicted values and the actual values of the stock.

For the aforementioned experiment the mean absolute percentage error (MAPE) = 2.01, and $R^2 = 0.87498$.

4. 1. Stock price forecasts for some airlines

We have trained four HMM for four different Airlines stock price. Using the same training dataset we trained four different (same architecture) ANN. Then we predicted the next few day's closing price of these three stocks using the HMMs in aforementioned method and we predicted the same day's closing price using ANN. The Table 2 shows the information of the training dataset and the test dataset while the Table 3 shows the prediction accuracy of these two models.

Table 2. Information on training and test datasets

Stock Name	Training Data		Test Data	
	From	To	From	To
British Airlines	17/09/2002	10/09/2004	11/09/2004	20/01/2005
Delta Airlines	27/12/2002	31/08/2004	01/09/2004	17/11/2004
Southwest Airlines	18/12/2002	23/07/2004	24/07/2004	17/11/2004
Ryanair Holdings Ltd.	06/05/2003	06/12/2004	07/12/2004	17/03/2005

Table 3. Prediction accuracy of ANN and the proposed method

	ANN (MAPE)	Proposed method (MAPE)
British Airlines	2.283	2.629
Delta Airlines	9.147	6.850
Southwest Airlines	1.673	2.011
Ryanair Holdings Ltd.	1.492	1.928

5. Conclusion

ANN is well researched and established method that has been successfully used to predict time series behaviour from past datasets. In this paper, we proposed the use of HMM, a new approach, to predict unknown value in a time series (stock market). It is clear from Table 3 that the mean absolute percentage errors (MAPE) values of the two methods are quite similar. Whilst, the primary weakness with ANNs is the inability to properly explain the models. According to Repley "the design and learning for feed-forward networks are

hard". Judd [18] and Blum and Rivest [19] showed this problem to be NP- complete. The proposed method using HMM to forecast stock price is explainable and has solid statistical foundation. The results show potential of using HMM for time series prediction. In our future work we plan to develop hybrid systems using AI paradigms with HMM to further improve accuracy and efficiency of our forecasts.

6. References

- [1] Kuo R J, Lee L C and Lee C F (1996), Integration of Artificial NN and Fuzzy Delphi for Stock market forecasting, IEEE International Conference on Systems, Man, and Cybernetics, Vol. 2, pp. 1073-1078.
- [2] Kimoto T, Asakawa K, Yoda M and Takeoka M (1990), Stock market prediction system with modular neural networks, Proc. International Joint Conference on Neural Networks, San Diego, Vol. 1, pp. 1-6.
- [3] White H (1998), Economic Prediction Using Neural Networks: The Case of IBM Daily Stock Returns, Proceedings of the Second Annual IEEE Conference on Neural Networks, Vol. 2, pp. 451-458.
- [4] Chiang W C, Urban T L and Baldrige G W (1996), A Neural Network Approach to Mutual Fund Net Asset Value Forecasting, Omega, Vol. 24 (2), pp. 205-215.
- [5] Kim S H and Chun S H (1998), Graded forecasting using an array of bipolar predictions: application of probabilistic neural networks to a stock market index. International Journal of Forecasting, Vol. 14, pp. 323-337.
- [6] Romahi Y and Shen Q (2000), Dynamic Financial Forecasting with Automatically Induced Fuzzy Associations, Proceedings of the 9th international conference on Fuzzy systems, pp. 493-498.
- [7] Thammano A (1999), Neuro-fuzzy Model for Stock Market Prediction, Proceedings of the Artificial Neural Networks in Engineering Conference, ASME Press, New York, pp. 587-591.
- [8] Abraham A, Nath B and Mahanti P K (2001), Hybrid Intelligent Systems for Stock Market Analysis, Proceedings of the International Conference on Computational Science. Springer, pp. 337-345.
- [9] Raposo R De C T and Cruz A J De O (2004), Stock Market prediction based on fundamental analysis with Fuzzy-Neural Networks.
http://www.labic.nce.ufrj.br/downloads/3wses_fsfs_2002.pdf
- [10] Cao L and Tay F E H (2001), Financial Forecasting Using Support Vector Machines, Neural Computation and Application, Vol. 10, pp. 184-192.
- [11] Huang X, Ariki Y, Jack M (1990), Hidden Markov Models for speech recognition. Edinburgh University Press.
- [12] Jelinek F, Kaufmann M, Mateo C S (1990), Self-organized language modelling for speech recognition, in Readings in Speech Recognition (Eds. Alex Waibel and Kai-Fu Lee), Morgan Kaufmann, San Mateo, California, pp. 450-506.
- [13] Xie H, Anreae P, Zhang M, Warren P (2004), Learning Models for English Speech Recognition, Proceedings of the 27th Conference on Australasian Computer Science, pp. 323-329.
- [14] Liebert M A (2004), Use of runs statistics for pattern recognition in genomic DNA sequences. Journal of Computational Biology, Vol. 11, pp. 107-124.
- [15] Vinciarelli A and Luetttin J (2000), Off-line cursive script recognition based on continuous density HMM, Proceedings of the 7th International Workshop on Frontiers in Handwriting Recognition, Amsterdam, pp. 493-498.
- [16] Li Z, Wu Z, He Y and Fulei C (2005), Hidden Markov model-based fault diagnostics method in speed-up and speed-down process for rotating machinery. Mechanical Systems and Signal Processing, Vol. 19(2), pp. 329-339.
- [17] Rabiner R L (1989), A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, Vol. 77(2), pp. 257-286.
- [18] Judd J S, (1990), Neural Network design and Complexity of Learning. MIT Press, USA.
- [19] Blum A L and Rivest R L, (1992), Training a 3-node Neural Networks is NP- complete. Neural Networks, Vol. 5, pp. 117-127.